

NETMANAGEIT

Intelligence Report BlackCat Climbs the Summit With a New Tactic

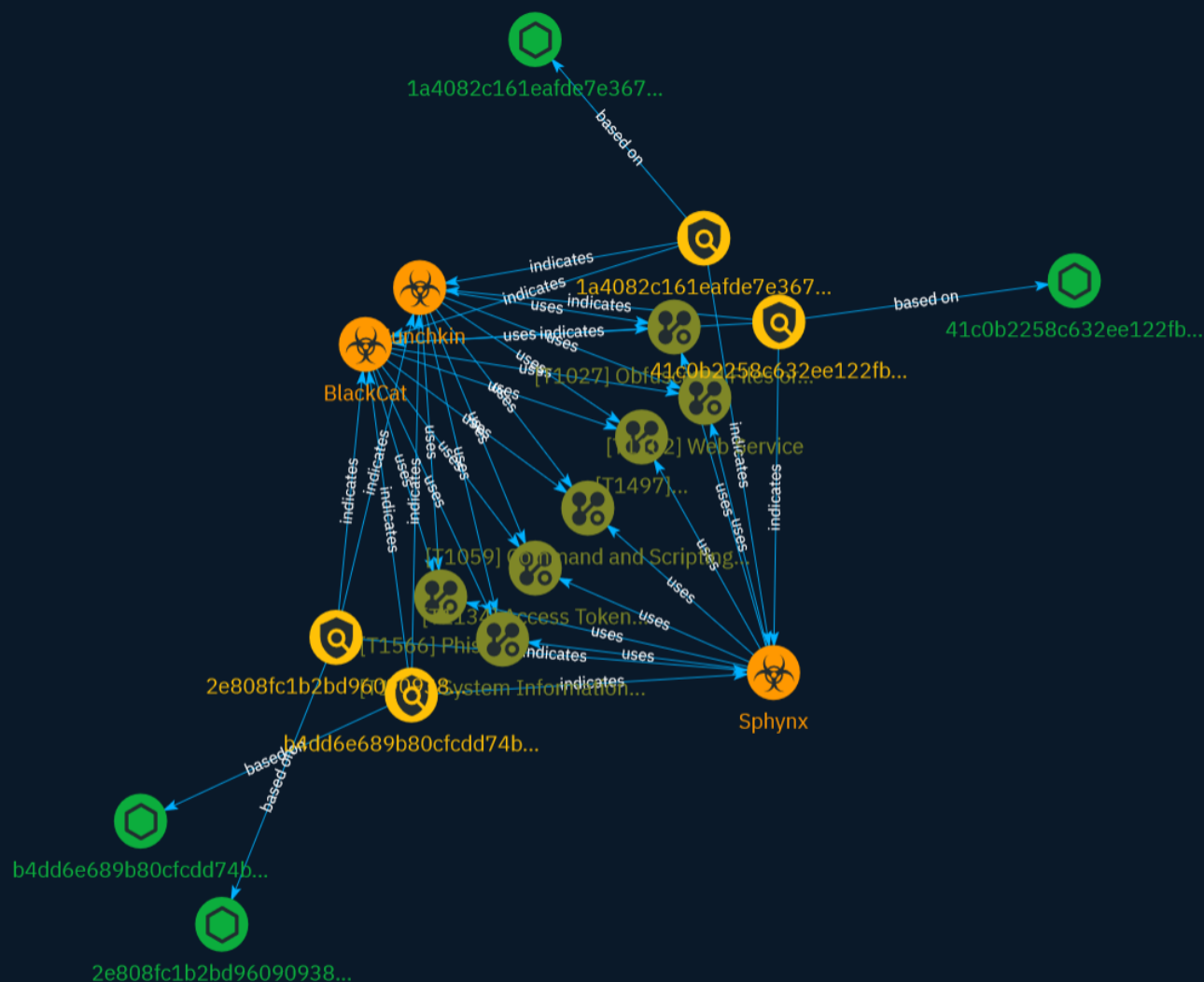


Table of contents

Overview

● Description	4
● Confidence	4
● Content	5

Entities

● Attack-Pattern	6
● Indicator	12
● Malware	14

Observables

● StixFile	15
------------	----



External References

- External References

16

Overview

Description

Unit 42 researchers have uncovered a new tooling used by the BlackCat ransomware threat group to deploy malware on remote machines and shares on a victim's network. the latest version of the malware family has been released.

Confidence

This value represents the confidence in the correctness of the data contained within this report.

15 / 100

Content

N/A

Attack-Pattern

Name

Virtualization/Sandbox Evasion

ID

T1497

Description

Adversaries may employ various means to detect and avoid virtualization and analysis environments. This may include changing behaviors based on the results of checks for the presence of artifacts indicative of a virtual machine environment (VME) or sandbox. If the adversary detects a VME, they may alter their malware to disengage from the victim or conceal the core functions of the implant. They may also search for VME artifacts before dropping secondary or additional payloads. Adversaries may use the information learned from [Virtualization/Sandbox Evasion](<https://attack.mitre.org/techniques/T1497>) during automated discovery to shape follow-on behaviors.(Citation: Deloitte Environment Awareness) Adversaries may use several methods to accomplish [Virtualization/Sandbox Evasion](<https://attack.mitre.org/techniques/T1497>) such as checking for security monitoring tools (e.g., Sysinternals, Wireshark, etc.) or other system artifacts associated with analysis or virtualization. Adversaries may also check for legitimate user activity to help determine if it is in an analysis environment. Additional methods include use of sleep timers or loops within malware code to avoid operating within a temporary sandbox. (Citation: Unit 42 Pirpi July 2015)

Name

Phishing

ID

T1566

Description

Adversaries may send phishing messages to gain access to victim systems. All forms of phishing are electronically delivered social engineering. Phishing can be targeted, known as spearphishing. In spearphishing, a specific individual, company, or industry will be targeted by the adversary. More generally, adversaries can conduct non-targeted phishing, such as in mass malware spam campaigns. Adversaries may send victims emails containing malicious attachments or links, typically to execute malicious code on victim systems. Phishing may also be conducted via third-party services, like social media platforms. Phishing may also involve social engineering techniques, such as posing as a trusted source, as well as evasive techniques such as removing or manipulating emails or metadata/headers from compromised accounts being abused to send messages (e.g., [Email Hiding Rules](https://attack.mitre.org/techniques/T1564/008)).(Citation: Microsoft OAuth Spam 2022)(Citation: Palo Alto Unit 42 VBA Infostealer 2014) Another way to accomplish this is by forging or spoofing(Citation: Proofpoint-spoof) the identity of the sender which can be used to fool both the human recipient as well as automated security tools.(Citation: cyberproof-double-bounce) Victims may also receive phishing messages that instruct them to call a phone number where they are directed to visit a malicious URL, download malware,(Citation: sygnia Luna Month)(Citation: CISA Remote Monitoring and Management Software) or install adversary-accessible remote management tools onto their computer (i.e., [User Execution](https://attack.mitre.org/techniques/T1204)).(Citation: Unit42 Luna Moth)

Name

Obfuscated Files or Information

ID

T1027

Description

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit.

This is common behavior that can be used across different platforms and the network to evade defenses. Payloads may be compressed, archived, or encrypted in order to avoid detection. These payloads may be used during Initial Access or later to mitigate detection. Sometimes a user's action may be required to open and [Deobfuscate/Decode Files or Information](https://attack.mitre.org/techniques/T1140) for [User Execution](https://attack.mitre.org/techniques/T1204). The user may also be required to input a password to open a password protected compressed/encrypted file that was provided by the adversary. (Citation: Volexity PowerDuke November 2016) Adversaries may also use compressed or archived scripts, such as JavaScript. Portions of files can also be encoded to hide the plain-text strings that would otherwise help defenders with discovery. (Citation: Linux/Cdorked.A We Live Security Analysis) Payloads may also be split into separate, seemingly benign files that only reveal malicious functionality when reassembled. (Citation: Carbon Black Obfuscation Sept 2016) Adversaries may also abuse [Command Obfuscation](https://attack.mitre.org/techniques/T1027/010) to obscure commands executed from payloads or directly via [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059). Environment variables, aliases, characters, and other platform/language specific semantics can be used to evade signature based detections and application control mechanisms. (Citation: FireEye Obfuscation June 2017) (Citation: FireEye Revoke-Obfuscation July 2017)(Citation: PaloAlto EncodedCommand March 2017)

Name

Access Token Manipulation

ID

T1134

Description

Adversaries may modify access tokens to operate under a different user or system security context to perform actions and bypass access controls. Windows uses access tokens to determine the ownership of a running process. A user can manipulate access tokens to make a running process appear as though it is the child of a different process or belongs to someone other than the user that started the process. When this occurs, the process also takes on the security context associated with the new token. An adversary can use built-in Windows API functions to copy access tokens from existing processes; this is known as token stealing. These token can then be applied to an existing process (i.e. [Token Impersonation/Theft](https://attack.mitre.org/techniques/T1134/001)) or used to spawn a new process (i.e. [Create Process with Token](https://attack.mitre.org/techniques/T1134/002)). An adversary must already be in a privileged user context (i.e. administrator)

to steal a token. However, adversaries commonly use token stealing to elevate their security context from the administrator level to the SYSTEM level. An adversary can then use a token to authenticate to a remote system as the account for that token if the account has appropriate permissions on the remote system.(Citation: Pentestlab Token Manipulation) Any standard user can use the ``runas`` command, and the Windows API functions, to create impersonation tokens; it does not require access to an administrator account. There are also other mechanisms, such as Active Directory fields, that can be used to modify access tokens.

Name

Command and Scripting Interpreter

ID

T1059

Description

Adversaries may abuse command and script interpreters to execute commands, scripts, or binaries. These interfaces and languages provide ways of interacting with computer systems and are a common feature across many different platforms. Most systems come with some built-in command-line interface and scripting capabilities, for example, macOS and Linux distributions include some flavor of [Unix Shell](<https://attack.mitre.org/techniques/T1059/004>) while Windows installations include the [Windows Command Shell](<https://attack.mitre.org/techniques/T1059/003>) and [PowerShell](<https://attack.mitre.org/techniques/T1059/001>). There are also cross-platform interpreters such as [Python](<https://attack.mitre.org/techniques/T1059/006>), as well as those commonly associated with client applications such as [JavaScript](<https://attack.mitre.org/techniques/T1059/007>) and [Visual Basic](<https://attack.mitre.org/techniques/T1059/005>). Adversaries may abuse these technologies in various ways as a means of executing arbitrary commands. Commands and scripts can be embedded in [Initial Access](<https://attack.mitre.org/tactics/TA0001>) payloads delivered to victims as lure documents or as secondary payloads downloaded from an existing C2. Adversaries may also execute commands through interactive terminals/shells, as well as utilize various [Remote Services](<https://attack.mitre.org/techniques/T1021>) in order to achieve remote Execution. (Citation: Powershell Remote Commands)(Citation: Cisco IOS Software Integrity Assurance - Command History)(Citation: Remote Shell Execution in Python)

Name

Web Service

ID

T1102

Description

Adversaries may use an existing, legitimate external Web service as a means for relaying data to/from a compromised system. Popular websites and social media acting as a mechanism for C2 may give a significant amount of cover due to the likelihood that hosts within a network are already communicating with them prior to a compromise. Using common services, such as those offered by Google or Twitter, makes it easier for adversaries to hide in expected noise. Web service providers commonly use SSL/TLS encryption, giving adversaries an added level of protection. Use of Web services may also protect back-end C2 infrastructure from discovery through malware binary analysis while also enabling operational resiliency (since this infrastructure may be dynamically changed).

Name

System Information Discovery

ID

T1082

Description

An adversary may attempt to get detailed information about the operating system and hardware, including version, patches, hotfixes, service packs, and architecture. Adversaries may use the information from [System Information Discovery](<https://attack.mitre.org/techniques/T1082>) during automated discovery to shape follow-on behaviors, including whether or not the adversary fully infects the target and/or attempts specific actions. Tools such as [Systeminfo](<https://attack.mitre.org/software/S0096>) can be used to gather detailed system information. If running with privileged access, a breakdown of system data can be gathered through the `systemsetup` configuration tool on macOS. As an example, adversaries with user-level access can execute the df -aH` command to obtain currently`

mounted disks and associated freely available space. Adversaries may also leverage a [Network Device CLI](<https://attack.mitre.org/techniques/T1059/008>) on network devices to gather detailed system information (e.g. `show version`).(Citation: US-CERT-TA18-106A) [System Information Discovery](<https://attack.mitre.org/techniques/T1082>) combined with information gathered from other forms of discovery and reconnaissance can drive payload development and concealment.(Citation: OSX.FairyTale)(Citation: 20 macOS Common Tools and Techniques) Infrastructure as a Service (IaaS) cloud providers such as AWS, GCP, and Azure allow access to instance and virtual machine information via APIs. Successful authenticated API calls can return data such as the operating system platform and status of a particular instance or the model view of a virtual machine.(Citation: Amazon Describe Instance)(Citation: Google Instances Resource)(Citation: Microsoft Virtual Machine API)

Indicator

Name

b4dd6e689b80cfcdd74b0995250d63d76ab789f1315af7fe326122540cddfad2

Pattern Type

stix

Pattern

[file:hashes!'SHA-256' =
'b4dd6e689b80cfcdd74b0995250d63d76ab789f1315af7fe326122540cddfad2']

Name

41c0b2258c632ee122fb52bf2f644c7fb595a5beaec71527e2ebce7183644db2

Pattern Type

stix

Pattern

[file:hashes!'SHA-256' =
'41c0b2258c632ee122fb52bf2f644c7fb595a5beaec71527e2ebce7183644db2']

Name

1a4082c161eafde7e367e0ea2c98543c06dce667b547881455d1984037a90e7d

Pattern Type

stix

Pattern

[file:hashes!'SHA-256' =
'1a4082c161eafde7e367e0ea2c98543c06dce667b547881455d1984037a90e7d']

Name

2e808fc1b2bd960909385575fa9227928ca25c8665d3ce5ad986b03679dace90

Pattern Type

stix

Pattern

[file:hashes!'SHA-256' =
'2e808fc1b2bd960909385575fa9227928ca25c8665d3ce5ad986b03679dace90']

Malware

Name

Munchkin

Name

Sphinx

Name

BlackCat

Description

[BlackCat](<https://attack.mitre.org/software/S1068>) is ransomware written in Rust that has been offered via the Ransomware-as-a-Service (RaaS) model. First observed November 2021, [BlackCat](<https://attack.mitre.org/software/S1068>) has been used to target multiple sectors and organizations in various countries and regions in Africa, the Americas, Asia, Australia, and Europe.(Citation: Microsoft BlackCat Jun 2022)(Citation: Sophos BlackCat Jul 2022)(Citation: ACSC BlackCat Apr 2022)

StixFile

Value

b4dd6e689b80cfcdd74b0995250d63d76ab789f1315af7fe326122540cddfad2

41c0b2258c632ee122fb52bf2f644c7fb595a5beaec71527e2ebce7183644db2

2e808fc1b2bd960909385575fa9227928ca25c8665d3ce5ad986b03679dace90

1a4082c161eafde7e367e0ea2c98543c06dce667b547881455d1984037a90e7d

External References

-
- <https://otx.alienvault.com/pulse/6532fad4104f6577740b7a15>
-
- <https://unit42.paloaltonetworks.com/blackcat-ransomware-releases-new-utility-munchkin/>